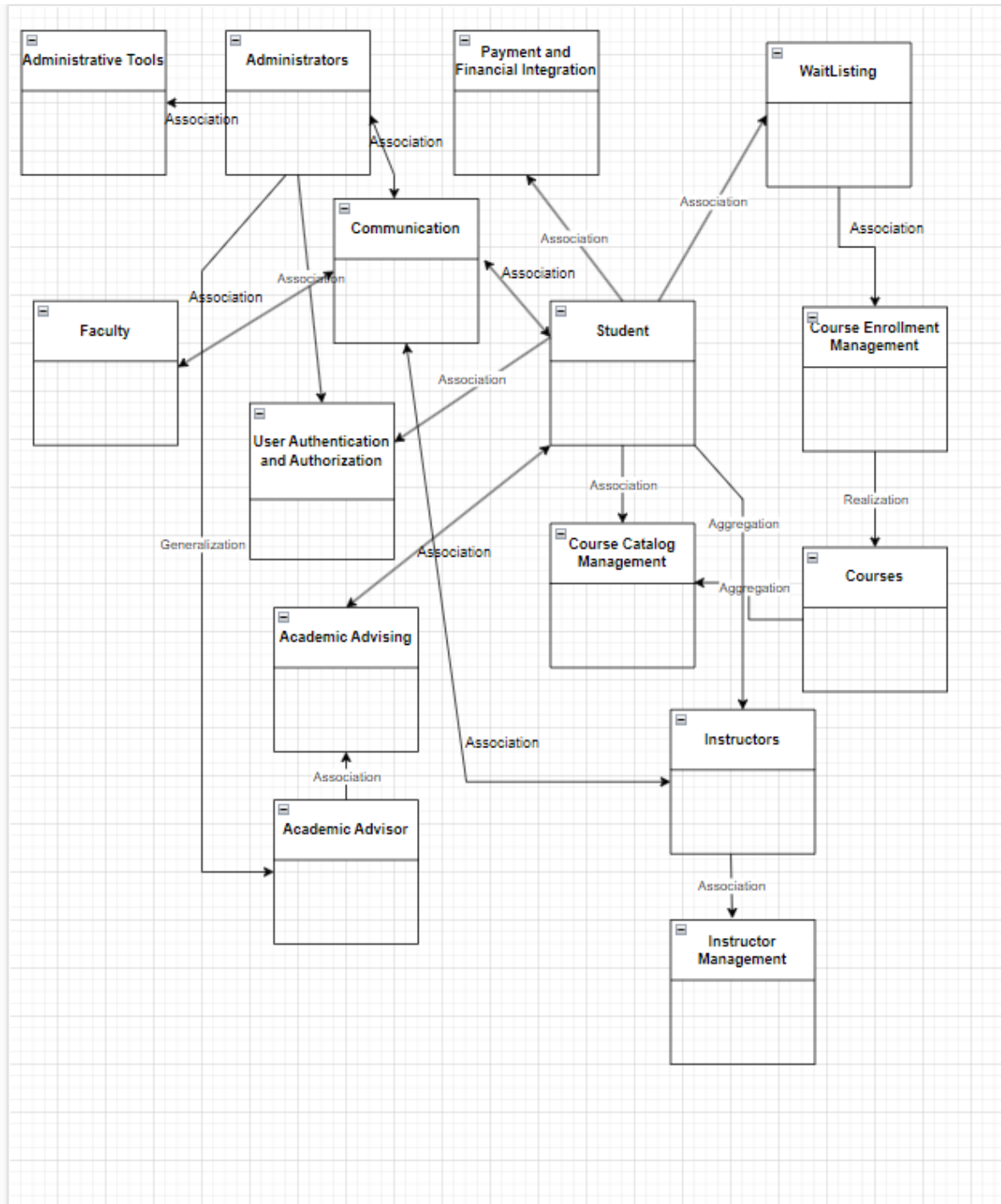


Q1.



Q2.

1. **The SRP (Single Responsibility Principle)**

Class CourseCatalog handles adding courses to the course list for the student and handles waitlisting of the course. The module can be changed for the following reasons: 1) Waitlisting format changes and 2) Content of the course list requirement changes. Since no module should have more than one reason to be changed, it's violating the SRP. These responsibilities should be redistributed in to two modules.

2. **The OPC (Open-Closed Principle)**

Class Prerequisites are taking in Course class object as a parameter for its constructor. Instead, Prerequisites class can inherit Course class allowing for extension and prohibiting modification. This will keep the reusability and maintainability high.

3. **The LSP (The Liscov Substitution Principle)**

Class Course is violating LSP through overriding hashCode, equals, and toString function. According to the SOLID principles, child class that inherits parent class must behave the same. In this case, it would be wise to create a separate function for these cases instead of overriding.

Q3.

- Class "Transcript" has 1 to 1 composition relationship with the class Student. Class Student includes Transcript as part of the class and Transcript class object will be destroyed once class student is destroyed.
- Class "Course" has 1 to many association relationship with class Student. Course class is used in student class but is not owned by the student class nor inherits the class.
- Class "Grade" has 1 to many association relationship with class Student. Grade class has some association with student class, but just by means to pass it to the Transcript class which has one to many aggregation class.